



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/645,438	08/21/2003	Tony McCormack	920673-94702	7992
23644 7590 09/15/2010 BARNES & THORNBURG LLP P.O. BOX 2786 CHICAGO, IL 60690-2786				
EXAMINER				
CHEA, PHILIP J				
ART UNIT		PAPER NUMBER		
2453				
NOTIFICATION DATE		DELIVERY MODE		
09/15/2010		ELECTRONIC		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

Patent-ch@btlaw.com

Office Action Summary

Application No.

10/645,438

Applicant(s)

MCCORMACK ET AL.

Examiner

PHILIP J. CHEA

Art Unit

2453

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 03 September 2010.
2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-3 and 5-17 is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5) ☐ Claim(s) _____ is/are allowed.
6) ☒ Claim(s) 1-3 and 5-17 is/are rejected.
7) ☐ Claim(s) _____ is/are objected to.
8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/GS/US)
Paper No(s)/Mail Date _____

- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date _____
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____

DETAILED ACTION

This Office Action is in response to an Amendment filed 9/3/10. Claims 1-3,5-17 are currently pending. Any rejection not set forth below has been overcome by the current Amendment.

Claim Rejections - 35 USC § 101

1. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 11-12,14-16 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 11 and 12 are directed to a program product comprising a computer-readable medium. Although the computer-readable medium mentions storing or recording instructions, one of ordinary skill in the art may construe the medium to include signals or carrier waves. In order to overcome this rejection, the Examiner suggests amending the claim to read "computer program product comprising a non-transitory computer-readable medium".

Claim 14 is directed to a computer-implemented system, but only positively recites software components of the system. A request handler, a network connection, comparison means, and routing instruction are considered software per se. In order to overcome this rejection, the Examiner suggests claiming a piece of hardware to perform the functions of those software components.

Claims 15 and 16 are directed to a software object that is a software data structure. The software data structure is considered software per se and does not fall under a statutory class of invention.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the

Art Unit: 2453

invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

3. Claims 1-3,5-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Flockhart et al. (US 6,535,601), and further in view of Webber (US 6,883,006).

As per claims 1,11,13, Flockhart discloses a method of contacts within a contact centre, comprising the steps of:

assigning a received contact a priority and a skillset identifier, whereby the received contact can be prioritized relative to other ones of the said contacts (see column 4, line 62 – column 5, line 5, *describing a received contact (i.e. caller) and priority levels access to tech support by supporting a value tag and a skillset identifier in the form of calls requiring a particular skill, such as technical support of various levels*);

creating a new software object for the received contact (see column 4, line 62 – column 5, line 5, *where the new software object is created for the call once entered into the queue, since the queue is a software data structure implemented by a computer see column 3, lines 49-56*);

determining a queuing position for said new software object relative to at least one other software object representing a contact having a skillset identifier similar to said skillset identifier assigned to said received contact (see column 4, line 62 - column 5, line 5, *where the contacts are placed into separate queues with varying tech support skills see Fig. 1A [121-123, 129], showing nine distinct queues related to distinct skills, for example, all contacts in Skill 1 Queue [121] have the same skillset identifier and column 5, lines 7-18, showing how the contact software objects in the queue have a queuing position relative to other contact software objects, implying that a new contact software objects position will be determined*);

adding to said new software object a pointer to said at least one other software object (see column 5, lines 7-18, *where the software objects in the queue have a pointer to other software objects in the form of queue position, that is the head position has a pointer to the next position all the way to the tail in order to determine the value of the calls see column 5, lines 43-46*);

storing in memory a collection of said software objects each containing said pointer to at least one other of said software objects (*i.e. the is implemented by a computer, implying that a memory is used to store the queue data structure*);

whereby said stored collection of said software objects provides a prioritized queue for a skillset (see column 5, lines 49-56, *showing a prioritized queue for a particular skillset, e.g. Skill 1 Queue or Skill 2 Queue*).

Although the system disclosed by Flockhart shows substantial features of the claimed invention (discussed above), it fails to disclose pointers when adding new software objects.

Nonetheless, these features are well known in the art and would have been an obvious modification of the system disclosed by Flockhart, as evidenced by Webber.

In an analogous art, Webber discloses a circular singly linked list having a number of list entries each of which has an associated next pointer field (see Abstract). Weber further disclose adding a new software object with a pointer to maintain a link to at least a second software object associated with another software object immediately ahead and/or before a software object in the queue (see column 3, lines 53-65, *showing how a software object (i.e. an entry with pointer and data field) is added to a queue*).

Given the teaching of Webber, a person having ordinary skill in the art would have readily recognized the desirability and advantages of modifying Flockhart by employing a linked list of software entries with pointers, such as disclosed by Webber, in order to add entries to the queue dynamically as the queue is required to grow.

In considering assigning multiple skillset identifiers and adding separate pointers to software objects in different queues, one of ordinary skill in the art would have had the knowledge at the time of the invention to place any arbitrary value in a linked list such as a list of different skillsets and create a pointer to an element in a list, whether it is the current list the element resides or if its another list such as for a multidimensional linked list. Evidence that shows knowledge of pointers and arbitrary elements can be found in prior art Goodrich et al. ("Lists and Sequences") provided in IDS filed 5/7/07. On page 2, the prior art shows a list of arbitrary objects and pointers to other objects. One of ordinary skill in the art could

Art Unit: 2453

see that a pointer can point to any object. It is obvious that the pointer can point to any other object to create a data structure such as a multidimensional linked list or tree.

As per claim 2, Flockhart in view of Webber further disclose that the new software object includes pointers (see Webber column 3, lines 53-65) to two other software objects having said similar skillset identifier, said two other software objects representing the contacts immediately ahead of and behind said contact within a queue, except the case in which the new software object is positioned at the end of said queue (see Flockhart Fig. 3, *showing a queue with a particular Tech Support Skill and a contact in the 4th position having contacts immediately ahead and behind the contact except when the contact is in the tail position e.g. 6th position*).

As per claim 3, Flockhart in view of Webber further disclose modifying said at least one other software object with a pointer (see Webber column 3, lines 53-65) to the new software object (*e.g. in Flockhart Fig. 3, when a new software object is entered into queue at the 7th position, the 6th position software object will have reference to the 7th position software object*).

As per claim 5, Flockhart further discloses responding to a network request by sending over a network details of those software objects at a head of a queue matching criteria specified in the request (see column 5, lines 46-56).

As per claim 6, Flockhart in view of Webber further disclose that the software objects are created and maintained by a contact manager, and a queuing module carries out said determination of said queuing position according to information associated with the new software object, the queuing module being further capable of adding said pointer (see Webber column 3, lines 53-65) to said new software object (see Flockhart Fig. 1A, *showing call vector [140] for adding contacts to the queue and Fig. 2 showing how the calls enter the queue in order of arrival so they are positioned in order of arrival and the reference is added because the call values are identified from the head back to the end of the queue*).

As per claim 7, Flockhart further discloses that the contact manager has a memory space in which said software objects are stored (see column 4, lines 17-22, *describing the call vector program assigning calls to queues implying that the calls are software objects being placed into queues since the*

Art Unit: 2453

queues are a software data structure), and the queuing module has a memory space in which said software objects are updated (see column 4, line 62 – column 5, line 5), and said memory spaces either form part of a common space (see column 3, lines 49-57).

As per claims 8,10,12,14, Flockhart in view of Webber disclose a method of distributing contacts across a network of contact centres, wherein each contact is represented by a software object maintained at one of said contact centres, each software object containing pointers (see Webber column 3, lines 53-65) to one or more other of said software objects maintained at the same contact centre to provide a queue of software objects at each said contact centre (see Flockhart Fig. 1A, *showing a contact centre with software objects (i.e. calls in queues) and Fig. 3 showing each software object having references to one or more software objects (i.e. head of queue having reference to next contact in the queue all the way to the tail)*), wherein the method comprises:

upon a network resource having the capability of handling said contacts with certain criteria becoming available, requesting from each said contact centre the highest priority queued software object matching said criteria (see Flockhart column 5, lines 43-56);

receiving information relating to each such highest priority queued software object from said contact centres (see Flockhart column 5, lines 43-56);

determining which software object represents the contact with the highest priority and/or best match for the available resource (see Flockhart column 5, lines 43-56); and

issuing the routing instructions to cause said contact to be routed to the resource (see Flockhart column 6, lines 22-28).

As per claims 9, Flockhart in view of Webber disclose wherein the contact centre which maintained the software object representing the selected contact carries out the further step of removing the selected software object from its queue and updating said software objects which contain said pointers (see Webber column 3, lines 53-65) to the selected software object, to thereby update the top of one or more said queues represented at said contact centre by a collection of said software objects (see Flockhart column 6, lines 22-28, *where the software object is moved from the queue to the call selection*

Art Unit: 2453

consideration pool and see Fig. 5A, where the head of the queue would be moved to the call selection pool implying that there will be a new head).

As per claim 15, Flockhart in view of Webber disclose a software object embodied on a computer-readable medium, representing a contact at a contact centre, said software object including: a pointer to one or more other of said software objects located immediately ahead of or behind said software object in a queue (see Webber column 3, lines 53-65), the software object further comprising an identifier to the contact which the software object represents and a skillset identifier enabling the software object to be identified in a search for software objects representing contact which match given skillset criteria (see Flockhart Fig. 3, *showing a queue with a particular Tech Support Skill and a contact software object in the 4th position having contacts immediately ahead and behind the contact*).

As per claim 16, Flockhart in view of Webber disclose a virtual queue of contact embodied on a computer-readable medium, wherein: each contact within the queue is represented by a software object including a pointer to one or more other said software objects located immediately ahead of or behind said software object in the queue (see Webber column 3, lines 53-65), the software object further comprising an identifier to the contact which the software object represents and a skillset identifier enabling the software object to be identified in a search for said software objects representing said contacts which match given skillset criteria, and wherein the order of said contacts within the queue is determinable from the aggregated references between said software objects (see Flockhart Fig. 3, *showing a queue with a particular Tech Support Skill and a contact software object in the 4th position having contacts immediately ahead and behind the contact and see column 5, lines 43-56, describing an order of contacts in the queue and how they are distributed to available agents that are compatible with the skillset*).

In considering separate pointers to software objects in different queues, one of ordinary skill in the art would have had the knowledge at the time of the invention create a pointer to an element in a list, whether it is the current list the element resides or if its another list such as for a multidimensional linked list. Evidence that shows knowledge of pointers and arbitrary elements can be found in prior art Goodrich

Art Unit: 2453

et al. ("Lists and Sequences") provided in IDS filed 5/7/07. On page 2, the prior art shows a list of arbitrary objects and pointers to other objects. One of ordinary skill in the art could see that a pointer can point to any object. It is obvious that the pointer can point to any other object to create a data structure such as a multidimensional linked list or tree.

4. Claim 17 is rejected under 35 U.S.C. 103(a) as being unpatentable over Flockhart in view of Webber as applied to claim 6 above, and further in view of Wood et al. (US 2004/0259531), herein referred to as Wood.

As per claim 17, Flockhart shows that the contact manager has a contact manager memory space and the queuing module has a queuing module memory space (see column 4, lines 17-22, *describing the contact manager as software implying a memory space* and column 4, line 62 – column 5, *describing the queuing module program implying a memory space*).

Although the system disclosed by Flockhart in view of Webber shows substantial features of the claimed invention (discussed above), it fails to disclose that each of said software objects is stored in two corresponding copies, a first of said copies being stored in the queuing module memory space and a second of said copies being stored in the contact manager memory space, and wherein a replication service is provided which is configured to ensure that changes to the first of said copies are reflected in corresponding changes to the second of said copies, and that changes to the second of said copies are reflected in corresponding changes to the first of said copies.

Nonetheless, these features are well known in the art and would have been an obvious modification of the system disclosed by Flockhart in view of Webber, as evidenced by Wood.

In an analogous art, Wood discloses routing messages and the need to minimize a single point of failure by replicating queues throughout a node (see paragraph 193). At the time of the invention, a person having ordinary skill in the art would have recognized the advantage of a replication service for the software object to ensure that changes to either a first or second copy would be replicated across the memory spaces, in order to provide service in the event of a failure.

Response to Arguments

5. Applicant's arguments filed 9/3/10 have been fully considered but they are not persuasive.

A) Applicant contends that Goodrich does not disclose an obvious reason to have a pointer point to any object and a multidimensional linked list or tree.

In considering A), the Examiner respectfully disagrees. It is old and well known in the art that linked lists are used as data structures and provide an advantage of easily adding an item to a list by dynamically changing it's size. It is also old and well known to use linked lists to create multidimensional arrays, trees, or other data structures that require linking the data elements easily. The pointers in the data structure can point to any arbitrary object. Therefore, it is obvious that any type of data structure can be reconstructed using the objects and pointers and that two lists could have reference pointers to each other.

B) Applicant contends that Flockhart, Webber and Goodrich do not disclose multiple skillset identifiers to a contact.

In considering B), the Examiner respectfully disagrees. Flockhart already discloses multiple skillsets in the form of queues of separate skillsets (see column 4, line 62 - column 5, line 5, *where the contacts are placed into separate queues with varying tech support skills see Fig. 1A [121-123, 129], showing nine distinct queues related to distinct skills, for example, all contacts in Skill 1 Queue [121] have the same skillset identifier*). At the time of the invention, one of ordinary skill in the art would have found it obvious that the queue could contain multiple skillsets and identify the multiple skillsets to a contact since there is already the available feature of separate queues with multiple skillsets. Putting multiple skillsets in a single queue would be an obvious modification to Flockhart in case a contact needed help with varying levels of skillsets.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to PHILIP J. CHEA whose telephone number is (571)272-3951. The examiner can normally be reached on M-F 6:30-4:00 (1st Friday Off).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Joseph Thomas can be reached on 571-272-6776. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Philip J Chea
Primary Examiner
Art Unit 2453

/Philip J Chea/
Primary Examiner, Art Unit 2453
9/8/10